**Software Requirements Specification**

**for**

**Course Auditor**

**Version 1.0 approved**

**Prepared by Team Beaver Fever**

**CMU**

**2/13/19**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

**4. System Features 4**

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

**5. Other Nonfunctional Requirements 4**

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

**6. Other Requirements 5**

**Appendix A: Glossary 5**

**Appendix B: Analysis Models 5**

**Appendix C: To Be Determined List 6**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Team | 2/13/19 | Initial editing | 1.0.0 |
| Team | 2/18/19 | Group edit | 1.0.1 |
| Team | 2/20/19 | Group edit | 1.0.2 |
| Team | 2/22/19 | Group edit | 1.0.3 |
| Team | 2/25/19 | Group edit | 1.0.4 |
| Team | 4/23/19 | Group Edit | 1.0.5 |
|  |  |  |  |
|  |  |  |  |

# **Introduction**

## **Purpose**

The product specified in this document is a university student and course auditor developed by Team Beaver Fever©. This SRS covers the entire scope of the project.

## **Document Conventions**

The application functionality is meant to be easily understood. Therefore, the only requirements you should absolutely prioritize are the sections on functionality. The listing and explanation of the functions of the application is all that is required to understand how the application will work.

## **Intended Audience and Reading Suggestions**

The intended audience for this application is for administrators. The SRS outlines the functionality of the application, and the reader should be able to understand how the application works by reading the sections below. However, full understanding of the inner workings of the application is not necessary for basic use of the application. Usage is meant to be intuitive and simple. Queries of a certain student display their pertinent information. Queries of certain classes display prerequisites and courses that require the queried course. Full reading of this document is not necessary to understand those concepts.

## **Product Scope**

The purpose of this software is to provide administrators way to access student information and relevant coursework. They will also be able to identify courses that are still required for graduation. One of the main goals is to integrate an intuitive interface.

## **References**

None.

# **Overall Description**

## **Product Perspective**

This product is a new, completely self contained system. It is a simple model using a database of student information which is pulled and displayed to the specific student.

## **Product Functions**

* Contain database of students
* Search database based on name, course number, course roster, or courses taken
* Calculate basic statistics for students including:
  + Checks if the student has under 24 credits
  + Average graduation rate (in years or semesters)
  + Expected number of classes till graduation
  + Average GPA for the required classes
  + Time that a student has been attending the school

## **User Classes and Characteristics**

There are two major user classes that we anticipate will use this product immediately. These user classes are:

* Administrator: (This class will not be finished in the initial release) This user class will use the product to search for the needed information relating to a student within the system. Administrators have the access rights to gain information on any student available in the system.
* Student: The student user class will only be able to access information from the database that is relevant to their own personal account.

Both user classes will be able to access the same information layout. Although, the administrator is the only user that can look at multiple students, a student can only look at their own information. Because of this the administrator user has a higher security/privilege clearance. Experience will not prohibit any user from the usability of the product, as it is efficient and simple to use. Both user classes will have a username and password which gives them the correct level of access to the product once verified.

## **Operating Environment**

The software will operate in the Windows operating system. The final product will be a Executable file that will be a standalone product. No other applications will be needed for this software to run.

## **Design and Implementation Constraints**

Beaver Fever should not have any design or implementation restrictions for this project.

## **User Documentation**

Will be updated later.

## **Assumptions and Dependencies**

This program depends on a database that is formatted correctly, which is important to use the information (For example, the course number must have the season taken as the last three numbers). It is also assumed that there is a connection to the database, otherwise the program cannot perform its task.

# **External Interface Requirements**

## **User Interfaces**

No external user interfaces were used.

## **Hardware Interfaces**

The application we will be making will be supported to run on any computer that is running Windows operating system. The data will be built into the software, no external data storage is needed. Once the executable is run the user will have everything they need to use the product.

## **Software Interfaces**

A JDBC driver was used in order to communicate with the external SQL database.

## **Communications Interfaces**

No external interfaces were used.

# **System Features**

## Average GPA for Required Classes

4.1.1 Description and Priority

Medium Priority: It is important to see what the average GPA is for a students classes.

For Administrators, they can view the average GPA of all students in a particular class.

4.1.2 Stimulus/Response Sequences

Based on login information, the user will be able to view either the student or admin view.

From there the user will be able to view their information on GPA.

4.1.3 Functional Requirements

This function requires both functionality from the GUI and the user information. The GUI

will be displaying the information and the user information is needed in order to determine

what information is going to be displayed

REQ-1: GUI

REQ-2: User Information

## **System Feature 2 (and so on)**

…….To be continued

# **Other Nonfunctional Requirements**

## **Performance Requirements**

Any windows 10 system will be able to run the application easily. The data will not take much memory and will be stored within the software itself. There are no feature-specific requirements for this application, only that there is enough memory space for the application.

## **Safety Requirements**

There exists a possibility that the data organized in the application could be lost in the event of outside damage to the memory. Due to this, it is advisable for users to backup their data. The data of the application is stored by the software within itself. Therefore, data could be lost if memory and/or file damage occurs to any of the application’s files.

## **Security Requirements**

The audience of the application is educational administrators and students. Due to educational privacy laws, the course work of students is required to be protected and only accessible by privileged users. Administrators can only see relevant student information and data, while students can only view their own information and data.

## **Software Quality Attributes**

Due to the nature of auditing systems, accuracy is more important than accessibility. In the case of this project, the data is stored locally. Therefore there are no concerns regarding data integrity/accessibility that would be relevant were the application data stored using a cloud solution.

## **Business Rules**

Students can only read data, and can only read their own. Administrators are the only ones with write access but can also read any student relevant to their privilege level. These rules will require authentication upon the initialization of the application.

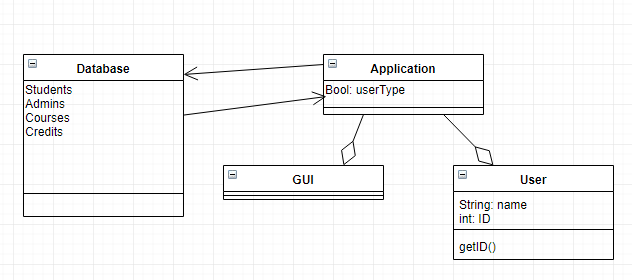
# **Other Requirements**

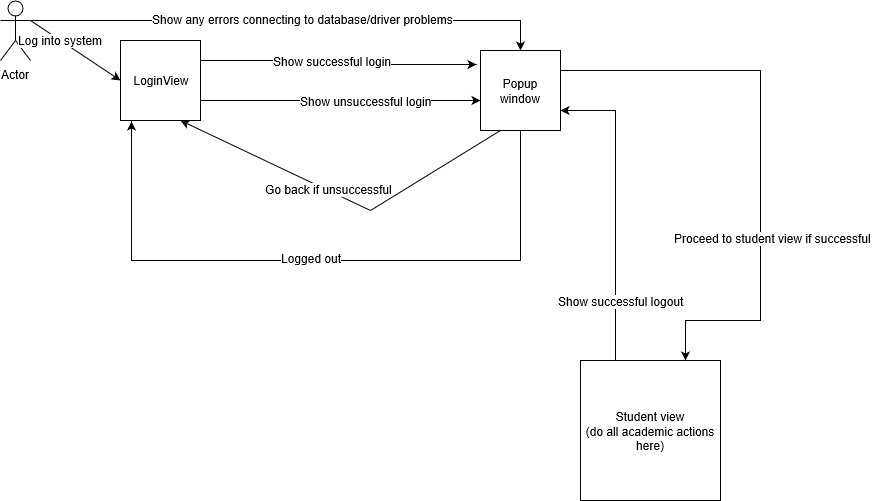
Beaver Fever will be making a simple database in order to store the data for the students. This database will interact with main application and be packaged directly.

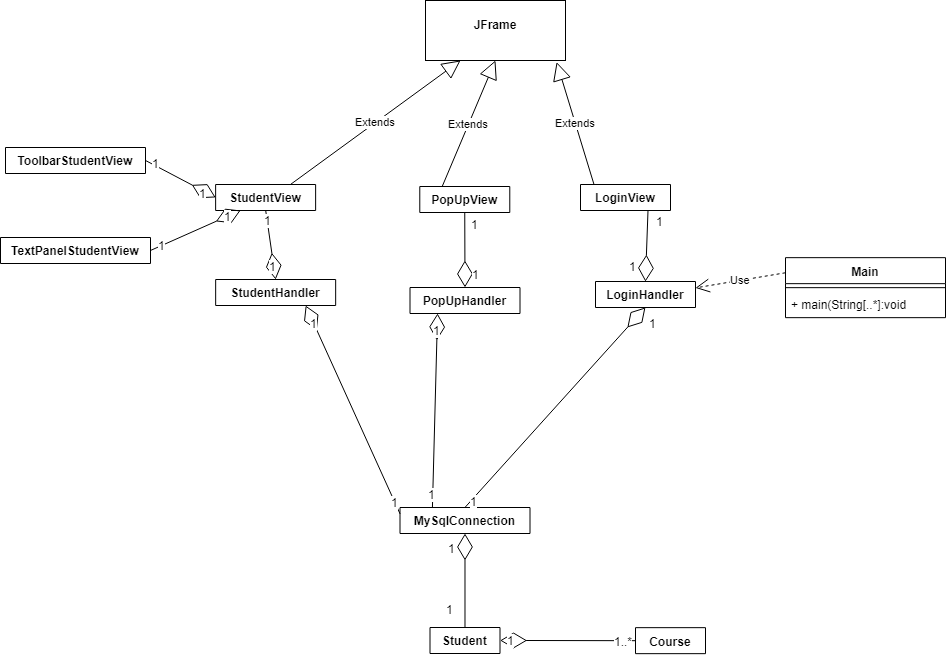
**Appendix A: Glossary**

*To be Continued*

**Appendix B: Analysis Models**

**

**

****

**Appendix C: To Be Determined List**

*To be Determined*